REMARKS

The examiner rejected Claims 23-32 under 35 U.S.C. 101 as being directed to non-statutory subject matter. The examiner stated:

> **The specification does not provide antecedent basis for "computer program product" recited in claims 23-29. Further, in view of claim 19, the "product" is software per se; therefore, claims 23-29 are directed to non-statutory subject matter as being intangible embodied.**
>
> **The language of claims 30-32 raise a question as to whether the claim is directed merely to an abstract idea that is not tied to a technological art, environment or machine which would result in a practical application producing a concrete, useful, and tangible result to form the basis of statutory subject matter under 35 U.S.C. 101. All of the recited steps of the method of claim 30-32 could be one by a person as a mental step or using pencil and paper. Claims 30-32 are directed to non-statutory subject matter.**

Applicant's specification provides ample support for "computer program product" recited in claims 23-29. See for instance, page 4, lines 25-30. See also page 11, lines 30-32: "The database 70 can be implemented using various approaches including hierarchical, relational or object oriented databases, or alternatively, a software or hardware cache."

The examiner also stated that: "Further, in view of claim 19, the "product" is software per se." Appellant disagrees. The preamble of claim 19 recites: "A computer program product residing on a computer readable medium for managing a cache for predicting availability information for a mode of transportation, comprises instructions to cause a computer to:" Claim 19 does not claim software per se. Rather, claim 19 claims a computer readable medium on which resides a computer program product. A computer readable medium is fully supported in Applicant's specification and is an article of manufacture under 35 U.S.C. 101.

Applicant has amended claim 23 to call for: "A computer program product residing on a computer readable medium for determining seat availability in a travel planning system, the computer program product comprising instructions to cause a computer to:." By reciting, that the computer program product resides on a computer readable medium, clearly makes claim 23 tangibly embodied as an article of manufacture, namely a computer readable medium.

Claim 32 recites: "A computer implemented method for managing availability information ... ." Claim 32 is thus tied to a technological art, which results in a practical

application, producing a concrete, useful, and tangible result thus claiming statutory subject matter under 35 U.S.C. 101, i.e., a computer implemented method. Claim 32 directed to a computer implemented method, and requiring *inter alia* "determining which entries to add, delete, or update in a cache" could not be performed as a mental step or using pencil and paper. Therefore, claims 30-32 are directed to statutory subject matter.

The examiner rejected claims 12, 17, 27, 30 under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.

Applicant has amended claims 12, 17, 27 and 30 to overcome this rejection.

The examiner rejected claims 1-3, 5-21, 23-32 under 35 U.S.C. 103(a) as being unpatentable over Mehovic (US 6,122,642 A) in view of Filepp et al. (US2003101 67307 A1), The examiner stated:

> **As per claims 1, 5, 19, 23 and 30, Mehovic substantially teaches the claimed invention including an airline computerize reservation system ("CRS") to provide flight and seat availability information (Col. 2 lines 5-20), Mehovic also teaches at Fig. 4 a cache (Fig. 4, element 20) stores data propagated from the CRS 12 data, which is used to response to queries from client 26 (Col. 3 lines 54-58).**
>
> **The different between Mehovic's system and the claimed invention is that Mehovic uses different cache management algorithm. Mehovic synchronizes the cache 20 with the CRS by propagating data immediately after CRS 12 updates the data or at definable intervals of time (Col. 3 lines 59-65), and therefore does not teach proactively update the cache based on frequency of access to the cache as claimed. However, Filepp teaches an airline reservation system (page 4, [0052]) utilizing caches storage (Fig. 2, 302) wherein the objects in caches are proactively updated based on frequency of access to the objects in the caches (page 50, [0821]-[0823]). Thus, it would have been obvious to one of ordinary skill in the art at the time of the invention was made to combine Filepp's cache management algorithm with Mehovic's CRS system so that "only the latest version of the object will be provided to guarantee currency of information to the user" as noted by Filepp at page 50, [0821]. By factoring the frequency of updating of updating of the objects in order to determine whether cached objects are current, Mehovic's system would detect the fights with high frequency of access, which implies that the number of available seats are also changed more frequently, and update the flight data so that the availability information for that flight is updated and current, therefore prevent overbooking or assigning the same seat to multiple passengers.**

Applicant's claims are allowable over Mehovic and Filepp et al. Without conceding that the examiner has provided a proper motivation to combine the teachings of the references, Applicant contends that the proposed combination fails to suggest the features of Applicant's claims.

Claim 1 for example calls for *** proactively determining if a stored answer in the cache is stale, the stored answer corresponding to seat availability information *** , with determining being based on a criterion for seat availability information, which criterion determined based on needs of a travel planning system, and if the stored answer *** is stale, sending an availability query to a source of seat availability information for the mode of transportation based on determining that the answer was stale. These features are not suggested by Mehovic and Filepp et al.

Mehovic teaches (Col. 2 lines 5-20):

> **The present invention provides a comprehensive system and methodology to automate the migration of data structures from an airline TPF based CRS processing environment and, in particular, the American Airlines' SABRE TPF based CRS environment to a relational database management system (RDBMS) for transparent retrieval and use by the end user. The invention thus provides a unique solution to a long standing data processing dilemma--how to propagate real-time airline TPF based CRS data quickly, directly, and transparently, from the airline TPF based CRS processing environment to any relational database management system (RDBMS) supported by Structured Query Language (SQL) command processing for use by the end user.**

Mehovic does not teach any of the features of claim 1. The examiner contends that Mehovic teaches to provide flight and seat availability information at (Col. 2 lines 5-20). However, this excerpt from Mehovic reproduced above, clearly shows that no such teaching exists in Mehovic.

The examiner also contends that Mehovic teaches (at Fig. 4 element 20) a cache that stores data propagated from the CRS 12 data, which is used to response to queries from client 26 (Col. 3 lines 54-58)." However, Mehovic deals with a fundamentally different problem than Applicant. While Applicant seeks to maintain fresh entries in a seat availability cache, e.g., in order to provide a reasonable prediction of seat availability before booking a ticket, Mehovic deals with a data translation problem, e.g., migrating data from TPF based CRS systems to RDBMS based end user systems. As such, Mehovic does not specifically deal with caching, and does not proactively determine if stored entries in the cache are stale. In particular, Mehovic does not determine staleness based on a criterion *** determined based on needs of a travel planning system that makes queries to the cache ***. Finally, nowhere does Mehovic teach

sending an availability query to a source of seat availability information *** based on determining that the answer was stale.

While the examiner seems to acknowledge this by saying that: "Mehovic uses different cache management algorithm." The examiner uses Filepp et al. to teach the "cache management algorithm." Filepp however, does not teach a staleness criterion based on needs of a travel planning system that makes queries to the cache. The examiner states that Filepp discloses: "wherein the objects in caches are proactively updated based on frequency of access to the objects in the caches (page 50, [0821]-[0823])." Claim 1 does not claim updating "based on frequency of access to the objects in the caches." Rather, Claim 1 requires that the criterion for updating the seat availability information is based on needs of the travel planning system that makes queries to the cache, not frequency of access to the objects in the cache.

Filepp, moreover, even fails to teach what the examiner relies on. The cited passages from Filepp are reproduced below:

> **[0821] When objects are requested from object storage facility 439, only the latest version of the object will be provided to guarantee currency of information to the user. Object storage facility 439 assures currency by requesting version verification from network 10 for those objects which are available locally and by requesting objects which are not locally available from delivery system 20 where currency is maintained.**
>
> **[0822] Version verification increases response time. Therefore, not all objects locally available are version checked each time they are requested. Typically, objects are checked only the first time they are requested during a user session. However, there are occasions, as for example in the case of objects relating to news applications, where currency is always checked to assure integrity of the information.**
>
> **[0823] The frequency with which the currency of objects is checked depends on factors such as the frequency of updating of the objects. For example, objects that are designated as ultrastable in a storage control parameter in the header of the object are never version checked unless a special version control object sent to the RS as part of logon indicates that all such objects must be version checked. Object storage facility 439 marks all object entries with such a stability category in all directories indicating that they must be version checked the next time they are requested.**

These passages deal with version testing to determine the frequency of checking versions of objects. These teachings have nothing at all to do with "wherein the objects in caches are proactively updated based on frequency of access to the objects in the caches" as contended by the examiner. Clearly these teaching have no relevance to updating the seat availability

information based on needs of a travel planning system that makes queries to the cache, as in claim 1.

According, Mehovic taken together with Filepp fails to suggest the features of claim 1.

Claim 2 further serves to distinguish over the combination of references. Claim 2 requires that determining if the stored answer is stale further comprises monitoring availability queries made to the cache by a travel planning system to determine which flights, sets of flights, the flights for a certain day, date, or market have a high demand for availability information.

The examiner states that Filepp et al. teach these limitations at [0821 to 0827]. Paragraphs [0821 to 0823] from Filepp were reproduced above, and paragraphs [0824 to 0827] from Filepp are reproduced below:

> **[0824] Object storage facility 439 manages objects locally in local store 440, comprised of a cache (segmented between available RAM and a fixed size disk file), and stage (fixed size disk file). Ram and disk cached objects are retained only during user sessions, while objects stored in the stage file are retained between sessions. The storage control field, located in the header portion of an object, described more fully hereafter as the object "storage candidacy", indicates whether the object is stageable, cacheable or trashable.**
>
> **[0825] Stageable objects must not be subject to frequent change or update. They are retained between user sessions on the system, provided storage space is available and the object has not discarded by a least-recently-used (LRU) algorithm of a conventional type; e.g., see Operating System Theory, by Coffman, Jr. and Denning, Prentice Hall Publishers, New York, 1973, which in accordance with the invention, operates in combination with the storage candidacy value to determine the object storage priority, thus rendering the stage self-configuring as described more fully hereafter. Over time, the self-configuring stage will have the effect of retaining within local disk storage those objects which the user has accessed most often. The objects retained locally are thus optimized to each individual user's usage of the applications in the system. Response time to such objects is optimized since they need not be retrieved from the interactive computer system.**
>
> **[0826] Cacheable objects can be retained during the current user session, but cannot be retained between sessions. These objects usually have a moderate update frequency. Object storage facility 439 retains objects in the cache according to the LRU storage retention algorithm. Object storage facility 439 uses the LRU algorithm to ensure that objects that are least frequently used forfeit their storage to objects that are more frequently used.**
>
> **[0827] Trashable objects can be retained only while the user is in the context of the partitioned application in which the object was requested. Trashable objects usually have a very high update frequency and must not be retained to ensure that the user has access to the most current data.**

Paragraphs [0821 to 0823] do not suggest the features of claim 2, since as discussed above, they deal 3with version testing of objects. Paragraphs [0824 to 0827] likewise do not suggest the features of monitoring availability queries made to the cache by a travel planning

system to determine which flights, sets of flights, the flights for a certain day, date, or market have a high demand for availability information. These paragraphs like [0821 to 0823] deal with criteria of the objects in the database, not criteria of a need of a system outside of the database. For instance, in claim 2, it is the determination of which flights, sets of flights, the flights for a certain day, date, or market have a high demand for availability information that is used as a criteria, not the stageability of the objects, the cacheability of the objects or the trashability of the objects, as taught by Filepp et al.

Claim 5 calls for a cache manager that manages a quality level of entry information in the cache by proactively populating the cache to maintain a high quality level of entries of seat availability information ..., with the quality level ... determined by evaluating entries in the cache according to a criterion related to needs of a travel planning system ... and that sends an availability query to a source of seat availability information for the mode of transportation based on determining that the seat availability information in the cache was stale. For similar reasons as discussed above, this claim distinguishes over Mehovic in view of Filepp et al.

Claim 19 calls for ... instructions to cause a computer to proactively determine whether a stored answer in the cache is stale, the stored answer corresponding to seat availability information ..., with instructions to determine being based on a ... criterion ... determined based on needs of a travel planning system .... The claim also recites instructions to update the stored answer in the cache when the stored answer is stale by sending an availability query to a source of availability information for the mode of transportation. For similar reasons, as discussed above, this claim is also allowable over Mehovic in view of Filepp et al.

Claim 23 calls for instructions to ... manage a quality level of entry information in the cache with the quality level of the seat availability information in the cache determined by comparing a criterion of the entries in the cache to criterion determined based on needs of a travel planning system that makes queries to the cache .... Claim 23 also recites deleting or modifying the entry based on determining that the entry should be deleted or modified, and proactively populating the cache by sending an availability query to a source of seat availability information *** based on determining that the entry for the seat availability information in the

cache should be added. For similar reasons as discussed above this claim is also allowable over Mehovic in view of Filepp et al.

Claim 30 recites ... determining, which entries to add, delete, or update in the cache by monitoring and examining availability queries made to the cache by a travel planning system to determine which instances of transportation have a high demand for availability information, and proactively updating entries in the cache if an instance is determined to have a higher than average or higher than expected demand.

Mehovic in view of Filepp et al neither describe nor suggest these features. Neither Mehovic nor Filepp et al. describe sending an actual availability query to a source of availability information. While of Filepp et al while discloses a cache, Mehovic in view of Filepp et al. does not suggest populating a cache based monitoring and examining availability queries made to the cache by a travel planning system to determine which instances have a high demand for availability information and updating entries in the cache based on if an instance is determined to have a higher than average or higher than expected demand. This element is not suggested by Mehovic in view of Filepp et al.

The examiner rejected claims 4 and 22 under 35 U.S.C. 103(a) as being unpatentable over Mehovic and Filepp as applied to claims above, and further in view of Khosravi-Sichani (US 5,983,217 A), hereinafter "Khosravi".

These claims are also allowable, since Mehovic and Filepp do not suggest the features of the base claims and Khosravi fails to teach the features of these claims.

The examiner stated:

> **As per claims 4, 22, Mehovic and Filepp teach the method of claims 1, 19 as discussed above. Mehovic and Filepp do not teach the step of processing query entry using round-robin algorithm as claimed. However, querying using round-robin is well know in the art, as exemplary by Khosravi. Khosravi teaches a method of querying replicate database using round-robin algorithm in order to "provide an even load sharing of queries" (Col. 1 lines 55-65). Thus, it would have been obvious to one of ordinary skill in the art at the time of the invention was made to combine Khosravi with Mehovic and Filepp's teaching because employing round-robin algorithm would ensure that all queries are processed equally and providing an even load sharing of queries.**

Claim 4 does not merely claim a round robin technique, but recites a specific approach that uses scheduling multiple lists, by processing one entry from each list by round-robin polling through the lists in turn until one entry has been processed from each list. However claim 4 also include the features of returning to the first list to process the next entry, generating an entry for each entry on the list in the order given, by submitting a query to the availability source; and storing the result in the cache, by updating an entry if present and adding an entry if not present in the cache.

Moreover, Khosravi teaches a technique for load balancing not a technique for determining if the stored answer is stale, as recited in claims 4 and 22.

Applicant's dependent claims add further distinct features to the invention as generally argued of record, and are at least allowable due to their dependency on allowable base claims.

The art cited but not applied is seen as neither describing nor suggesting applicant's invention whether taken separately or in combination with the art of record.

Accordingly, in view of the above amendments and remarks, the length of prosecution, and a prior indication of allowable subject matter, it is submitted that the case is now in condition for allowance and such action is respectfully requested.

Please apply any other charges or credits to deposit account 06-1050.

Respectfully submitted,

Date: 9/08/04

Denis G. Maloney
Reg. No. 29,670

Fish & Richardson P.C.
225 Franklin Street
Boston, MA 02110
Telephone: (617) 542-5070
Facsimile: (617) 542-8906